Optimal critic learning for robot control in time-varying environments

Article (Accepted Version)

Optimal Critic Learning for Robot Control in Time-Varying Environments

Chen Wang, Yanan Li, *Memeber, IEEE,* Shuzhi Sam Ge, *Fellow, IEEE,* and Tong Heng Lee, *Memeber, IEEE*

***Abstract*—In this paper, optimal critic learning is developed for robot control in a time-varying environment. The unknown environment is described as a linear system with time-varying parameters, and impedance control is employed for the interaction control. Desired impedance parameters are obtained in the sense of an optimal realization of the composite of trajectory tracking and force regulation. Q-function based critic learning is developed to determine the optimal impedance parameters without the knowledge of the system dynamics. Simulation results are presented and compared with existing methods, and the efficacy of the proposed method is verified.**

***Index Terms*—Interaction control, time-varying environment, critic learning, optimal control.**

I. Introduction

As the century unfolds, the application domain of robots has gradually expanded to human-inhabited environments, where the interaction control of robots has become increasingly challenging and important. There are several technical difficulties which need to be addressed as to develop an efficient and reliable interaction control: a) ensuring the safety of the robot and environment during the interaction; and b) realizing an adaptive behavior of the robot in spite of the change of environment parameters.

In the state-of-art of interaction control, there are two popular approaches, i.e., hybrid position/force control [1] and impedance control [2]. Compared to hybrid position/force control, impedance control is preferred as it does not require the direction decomposition and its robustness and feasibility have been widely acknowledged. The performance of impedance control relies on the proper selection of impedance parameters. In early research works, a set of desirable constant impedance parameters is usually prescribed and researchers' focus is how to deal with the uncertainties in the robot dynamics [3]. However, as robots are more expected to operate in unstructured and uncertain environments autonomously, conventional impedance control becomes too conservative to guarantee a good interaction performance since it is incapable of incorporating environment properties. To resolve this problem, impedance learning/adaptation with optimization is introduced in many research studies [4], [5], [6]. Optimization is important in impedance learning/adaptation since its objective includes both the trajectory tracking and force regulation. In [7], [8], the well-known Linear Quadratic Regulator (LQR) optimal control is adopted for the proper selection of impedance parameters. Although mathematically elegant, this approach has a major drawback posed by the requirement that the environment dynamics are completely known.

To design an impedance control for an unknown environment, estimation of environment parameters has been an option and was extensively studied in the literature [9], [10]. A desired impedance model can be constructed if the stiffness and damping parameters of the environment can be precisely estimated. In [9], a Recursive Least Square (RLS) scheme has been implemented to estimate the environment parameters. Unfortunately, as discussed in [11], identification is usually time-consuming because it requires the procedures of model design, parameter estimation, and model validation at each step of the iterations. In addition, considering the time-varying nature of most physical system models, those methods are not practical due to the relatively high computational requirements and slow response to parameter variations [12].

In order to derive a direct optimal control in the case of unknown system dynamics, Adaptive Dynamic Programming (ADP) or actor-critic learning is proposed [13], [14], [15], [16]. The idea of ADP is constructed by imitating the way human adapting to the surrounding environment. In particular, when performing a task, human can judge whether the action was successful, then the judgment is used to apply an update to the action [17], [18], [19], [20], [21], [22], [23], [24]. Under the structure of ADP, the control system is considered to include agents that are able to make decisions and modify their actions according to the environment stimuli. The action is strengthened or depressed according to the types of stimuli (positive reinforcement or negative reinforcement). Due to their unique critic-actor structure, optimal control can be generated with partial or no information of the system. There are existing works where ADP is successfully adopted for the impedance adaptation of robot control. In [25], [26], natural actor-critic algorithm is adopted and the damping and stiffness matrices are updated according to defined reward functions. As pointed out in [27], [28], [29], a learning process is still required in [25], [26] for the robot to repeat operations to learn the desired impedance parameters. To address this problem, in our previous work [6], impedance adaptation is introduced which does not require the repetitive learning process and thus provides a certain degree of convenience. However, as discussed in [30], any real physical system is time-varying, at least owing to the flicker noise in its components. The proposed method in [6] could not be applied to the scenario

C. Wang, S. S. Ge and T. H. Lee are with the Department of Electrical and Computer Engineering, and the Social Robotics Lab, Interactive and Digital Media Institute (IDMI), National University of Singapore, Singapore 117576 {wang_chen09,samge,eleleeth}@nus.edu.sg

Y. Li is with the Institute for Infocomm Research, Agency for Science, Technology and Research, Singapore 138632 liy@i2r.a-star.edu.sg

where the environment is time-varying due to the assumption that the environment dynamics are time-invariant.

In this paper, we focus on developing impedance adaptation in the case of unknown time-varying environments. In order to determine the optimal impedance parameters recursively online, optimal critic learning is developed for time-varying linear systems in discrete-time. In particular, the following problems will be addressed: a) the time-varying system parameters are not computationally tractable in practice; and b) when the system parameters change over time, optimized steady-state solutions may be too conservative in the sense that the execution of the optimal policy will be delayed and fail to handle such changes. The proposed method will take time-varying parameters into account, such that a different set of parameters and control policy are implemented for each adaptation step. The decision making and policy updating will require little computation cost, making the proposed method feasible in practical implementations. Compared to the previous work in [6], the problem under study is more challenging because developing an adaptive scheme usually requires a certain variable to be invariant, which is not satisfied in the case of a time-varying environment. The environment under study will be described as a linear system with unknown time-varying parameters. The developed impedance adaptation will result in desired impedance parameters that are able to guarantee the optimal interaction, subject to unknown and time-varying environments.

Based on the above discussions, we highlight the contributions of this paper as follows:

(i) The dynamics of unknown and time-varying environments are considered in the analysis of the interaction control problem, which are described as linear systems with unknown time-varying parameters.
(ii) Critic learning based on the recursive time-varying least square method is adopted to obtain the optimal control such that the online adaptation is achieved; and
(iii) Optimal impedance adaptation is developed in the sense of trajectory tracking and force regulation in the absence of unknown environment parameters.

The rest of the paper is organized as follows. In Section II, the environment dynamics are described, and the objective of this paper are discussed. In Section III, critic learning is developed for the described environment model, such that the optimal interaction is achieved subject to unknown time-varying environments. In Section IV, the validity of the proposed method is verified through simulation studies. Section V concludes this paper.

II. Problem Formulation

The system under study includes a rigid robot arm and an environment, and the end-effector of the robot arm physically interacts with the environment. A force sensor is mounted at the end-effector which can be used to measure the interaction force between the environment and the robot arm.

One common model used to define the normal component of the contact force is the spring-dashpot model [31]. In this model, the contact parameters (i.e., stiffness and damping) relate the position x to the normal force f at each contact point. Let k describe the time-step index, and the environment model in discrete-time is given as below

$$x(k+1) = A_e(k)x(k) + B_e(k)f(k) \tag{1}$$

where $A_e(k)$ and $B_e(k)$ are unknown time-varying matrices.

Remark 1: $A_e(k)$ and $B_e(k)$ are assumed to be unknown time-varying matrices in this paper. This assumption makes the problem studied in this paper more practical, yet more complicated compared with the previous studies in [8], [6].

Impedance control is first introduced in [2] to impose a desired dynamic behavior to the interaction between the robot and environment. To implement impedance control, we need to find a target impedance model in the Cartesian space as below

$$f(k) = \psi(x_d(k), x_r(k)) \tag{2}$$

where $x_d(k)$ is the given desired trajectory, $x_r(k)$ is the virtual desired trajectory in the Cartesian space, and $\psi(\cdot)$ is a target impedance function to be determined. Consider the robot arm kinematics as $x(k) = \phi(q(k))$, where $q(k) \in \mathbb{R}^n$ is the joint coordinates in the joint space. Then, the virtual desired trajectory in the joint space $q_r(k) = \phi^{-1}(x_r(k))$ can be determined according to the interaction force $f(k)$ and the impedance model (2).

The objective of this paper is to develop impedance adaptation which achieves optimal interaction performance for a robot system interacting with unknown time-varying environments. In particular, the control framework is shown in Fig. 1, which can be divided into two parts: a) optimal critic learning of impedance parameters and b) position control. In the first part, a proper impedance model $\psi(\cdot)$ needs to be found to achieve a certain optimal interaction performance. In order to realize this, the environment dynamics need be incorporated. However, as discussed in the Introduction, it is extremely difficult to identify the environment parameters when they are time-varying. In this regard, we aim to adopt the idea of optimal critic learning to determine the desired optimal impedance function, subject to an unknown and time-varying environment.

In the second part, position control (as shown in the dashed box in Fig. 1) is implemented to make $x(k) = x_r(k)$. The inner-loop is to guarantee the trajectory tracking, i.e., $\lim_{k\to\infty} q(k) = q_r(k)$. Trajectory tracking of a robot arm has been extensively studied in the literature [32], [33], so it will not be discussed in this paper. For the simplicity of analysis, it is assumed that there is an ideal inner-loop position control such that $q(k) = q_r(k)$ and thus $x(k) = x_r(k)$. In this way, the desired impedance model (2) becomes

$$f(k) = \psi(x_d(k), x(k)) \tag{3}$$

Based on the above discussion, the first part of the control structure is focused on in this paper, i.e., optimal critic learning of impedance parameters. This is non-trivial considering that $A_e(k)$ and $B_e(k)$ in the environment model (1) are unknown and time-varying. As discussed in the Introduction, iterative impedance learning and impedance adaptation have been de-

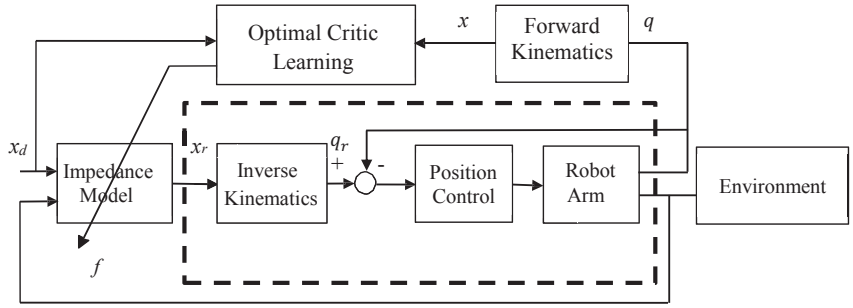

Fig. 1. Control framework

veloped in [6], [25], [26], [34], but very few methods have been developed for environments with time-varying parameters. This is the motivation to develop optimal impedance adaptation in the rest of this paper.

III. Control Design

A. Q-Function based Time-Varying LQR

In the following, we will formulate the time-varying LQR problem using the Bellman's principle of optimality and derive an online policy using the concept of Q-function [35], [36].

Consider the following Linear Time-Varying (LTV) system in the discrete-time domain

$$\xi(k+1) = A(k)\xi(k) + B(k)u(k) \tag{4}$$

where $\xi(k) \in \mathbb{R}^l$ is the system state, $u(k) \in \mathbb{R}^m$ is the system input, and $A(k)$ and $B(k)$ are time-varying matrices which are stabilizable.

The optimal control problem can be formulated by designing a control in the following form

$$u(k) = L(k)\xi(k) \tag{5}$$

which minimizes the below cost function

$$J = \sum_{k=1}^{\infty}[\xi^T(k)S\xi(k) + u^T(k)Ru(k)] \tag{6}$$

where $S \in \mathbb{R}^{l\times l}$ and $R \in \mathbb{R}^{m\times m}$ are weights of the state and input which satisfy $S = S^T \geq 0$ and $R = R^T > 0$, and $L(k)$ is the control gain.

In [37], heuristic dynamic programming (HDP) is developed to solve the following Discrete-Time Algebraic Riccati Equation (DARE)

$$\begin{aligned} P(k+1) &= A^T(k)P(k)A(k) + S \\ &\quad -A^T(k)P(k)B(k)[R + B^T(k) \\ &\quad \times P(k)B(k)]^{-1}B^T(k)P(k)A(k), \\ P(0) &= 0 \end{aligned} \tag{7}$$

where $P(k)$ is the solution of the DARE, which is in the feedback gain

$$\begin{aligned} L(k) &= -[R + B^T(k)P(k+1)B(k)]^{-1} \\ &\quad B^T(k)P(k+1)A(k) \end{aligned} \tag{8}$$

Remark 2: The above DARE can be also solved backward in time as below

$$\begin{aligned} P(k) &= A^T(k)P(k+1)A(k) + S \\ &\quad -A^T(k)P(k+1)B(k)[R + B^T(k) \\ &\quad \times P(k+1)B(k)]^{-1}B^T(k)P(k+1)A(k) \end{aligned} \tag{9}$$

with the terminal condition $P(\infty)$. Eqs. (7) and (9) produce the same sequence of $P(k)$, which converges to the solution of the DARE after enough iterations [38].

However, for this classical method, the system matrices $A(k)$ and $B(k)$ are assumed to be known beforehand so that the optimal solution is obtained recursively. As this condition is not satisfied in most cases, in the following, we will show how to derive the optimal solution based on critic learning which does not require prior information of system dynamics.

Consider the following infinite horizon cost-to-go function

$$V(\xi(k)) = \sum_{i=k}^{\infty}[\xi^T(i)S\xi(i) + u^T(i)Ru(i)] \tag{10}$$

The goal is to determine the optimal control policy

$$u^*(k) = \arg\min_{u(k)} V(\xi(k)) \tag{11}$$

Assuming that $u^*(k)$ exists, it is well-known that the corresponding cost-to-go function $V^*(\xi(k)) = \min_{u(k)} V(\xi(k))$ is quadratic in the state with the following form

$$V^*(\xi(k)) = \xi^T(k)P(k)\xi(k) \tag{12}$$

The cost-to-go function can be defined as

$$\begin{aligned} & V(\xi(k)) \\ = \; & g(\xi(k), u(k)) + V^*(\xi(k+1)) \\ = \; & \xi^T(k)S\xi(k) + u^T(k)Ru(k) + \\ & \xi^T(k+1)P(k+1)\xi(k+1) \\ = \; & \begin{bmatrix}\xi(k)\\ u(k)\end{bmatrix}^T \begin{bmatrix}S & 0\\ 0 & R\end{bmatrix} \begin{bmatrix}\xi(k)\\ u(k)\end{bmatrix} + \\ & \begin{bmatrix}\xi(k)\\ u(k)\end{bmatrix}^T \begin{bmatrix}A^T(k)\\ B^T(k)\end{bmatrix} P(k+1) \begin{bmatrix}A^T(k)\\ B^T(k)\end{bmatrix}^T \begin{bmatrix}\xi(k)\\ u(k)\end{bmatrix} \\ = \; & \begin{bmatrix}\xi(k)\\ u(k)\end{bmatrix}^T H(k) \begin{bmatrix}\xi(k)\\ u(k)\end{bmatrix} \end{aligned} \tag{13}$$

where $g(\xi(k),u(k)) = \xi^T(k)S\xi(k) + u^T(k)Ru(k)$ is the utility function at the k-th step. $H(k)$ in Eq. (13) can be further written as

$$H(k) = \begin{bmatrix} H_{\xi\xi} & H_{\xi u} \\ H_{u\xi} & H_{uu} \end{bmatrix} \tag{14}$$

where

$$\begin{aligned} H_{\xi\xi} &= A^T(k)P(k+1)A(k) + S \\ H_{\xi u} &= H_{u\xi}^T = A^T(k)P(k+1)B(k) \\ H_{uu} &= B^T(k)P(k+1)B(k) + R \end{aligned} \tag{15}$$

The optimal control policy can be acquired by

$$u(k) = L(k)\xi(k) = -\frac{\partial V(\xi(k))}{\partial u(k)} = -H_{uu}^{-1}H_{u\xi}\xi(k) \tag{16}$$

Eqs. (14) and (16) are the main equations needed to obtain the optimal control policy. Note that if $H(k)$ can be obtained using an online identification method, the system dynamics will no longer be needed. In the following, we will show how to formulate the optimal control problem using the Q-function based optimal principle, which will be further used to approximate the solution of the DARE in Eq. (7).

When the control policy is optimal, Eq. (13) is equal to Eq. (12). Therefore, the relationship between $P(k)$ and $H(k)$ can be obtained by

$$\begin{bmatrix} \xi(k) \\ u(k) \end{bmatrix}^T H(k) \begin{bmatrix} \xi(k) \\ u(k) \end{bmatrix} = \xi^T(k)P(k)\xi(k) \tag{17}$$

Noticing that $u(k) = L(k)\xi(k)$, the relationship between $H(k)$ and $P(k)$ can be obtained as

$$P(k) = \begin{bmatrix} I & L^T(k) \end{bmatrix} H(k) \begin{bmatrix} I & L^T(k) \end{bmatrix}^T \tag{18}$$

Let us define the following state and action based Q-function

$$Q(\xi(k),u(k)) = V(\xi(k)) = \begin{bmatrix} \xi(k) \\ u(k) \end{bmatrix}^T H(k) \begin{bmatrix} \xi(k) \\ u(k) \end{bmatrix} \tag{19}$$

The optimal control problem described in Eq. (11) then becomes finding the optimal control policy $u^*(k)$, which satisfies the following time-varying temporal difference equation

$$\begin{aligned} & Q^*(\xi(k),u^*(k)) \\ = \; & g(\xi(k),u^*(k)) + Q^*(\xi(k+1),u^*(k+1)) \end{aligned} \tag{20}$$

Remark 3: For a discrete-time system, the Q-function can be constructed as in Eqs. (13) and (19). However, the Q-function is relatively difficult to construct for a continuous-time system as the cost-to-go function $V(\xi)$ in a continuous-time system cannot be approximated using a Q-function which is quadratic in ξ and u. This is a major barrier to a completely model-free continuous ADP and will be further investigated in the future work. In this regard, we only consider the discrete-time implementation of the optimal critic learning in this paper.

B. Optimal Critic Learning

In this section, we use the Q-function in Section III-A to develop optimal critic learning for the time-varying system. The key idea is employing the successive Q-learning approximation method to solve the Hamilton-Jacobian-Bellman (HJB) equation [39], which is summarized in Algorithm 1.

Algorithm 1 Q-learning Approximation

1: Choose a stable control policy $v_0(\xi(k))$ and let the iteration index $j = 0$.

2: **Policy Evaluation** Solve for Q_{j+1} from

$$\begin{aligned} & Q_{j+1}(\xi(k),u(k)) \\ = \; & g(\xi(k),u(k)) + Q_j(\xi(k+1),v_j(\xi(k+1))) \end{aligned} \tag{21}$$

where v_j is the control policy.

3: **Policy Improvement** Update the control policy

$$v_{j+1}(\xi(k)) = \underset{u(k)}{\operatorname{argmin}}(Q_{j+1}(\xi(k),u(k))) \tag{22}$$

4: Let $j \leftarrow j+1$ and go to Step 2.

Lemma 1: If the system (4) is stabilizable, by iterating on Eqs. (21) and (22), $Q_j(\xi(k),u(k))$ will converge to $Q^*(\xi(k),u^*(k))$, and $v_{j+1}(\xi(k)) = \bar{L}_{j+1}(k)\xi(k)$ where $\bar{L}_{j+1}(k)$ is the approximation of optimal control gain $L_{j+1}(k)$, will converge to $u^*(k)$ as $j \to \infty$.

Proof 1: From Eq. (19), we have

$$\begin{aligned} & Q_{j+1}(\xi(k),u(k)) \\ = \; & z^T(k)\bar{H}_{j+1}(k)z(k) \\ & Q_j(\xi(k+1),v_j(\xi(k+1))) \\ = \; & z^T(k+1)\bar{H}_j(k+1)z(k+1) \end{aligned} \tag{23}$$

where $z(k) = \begin{bmatrix} \xi^T(k) & u^T(k) \end{bmatrix}^T$, $z(k+1) = \begin{bmatrix} \xi^T(k+1) & \bar{L}_j(k+1)\xi(k+1) \end{bmatrix}^T$ and $\bar{H}_{j+1}(k)$ is the approximation of $H(k)$ at the $(j+1)$-th iteration. By substituting Eqs. (4) and (23) into (21), we obtain

$$\begin{aligned} & z^T(k)\bar{H}_{j+1}(k)z(k) \\ = \; & z^T(k)\begin{bmatrix} A(k) & B(k) \\ \bar{L}_j(k+1)A(k) & \bar{L}_j(k+1)B(k) \end{bmatrix}^T \bar{H}_j(k+1) \\ & \times \begin{bmatrix} A(k) & B(k) \\ \bar{L}_j(k+1)A(k) & \bar{L}_j(k+1)B(k) \end{bmatrix} z(k) \\ & + z^T(k)\begin{bmatrix} S & 0 \\ 0 & R \end{bmatrix} z(k) \end{aligned} \tag{24}$$

Then, it is easy to obtain

$$\begin{aligned} & \bar{H}_{j+1}(k) \\ = \; & \begin{bmatrix} S & 0 \\ 0 & R \end{bmatrix} + \begin{bmatrix} A(k) & B(k) \\ \bar{L}_j(k+1)A(k) & \bar{L}_j(k+1)B(k) \end{bmatrix}^T \\ & \times \bar{H}_j(k+1)\begin{bmatrix} A(k) & B(k) \\ \bar{L}_j(k+1)A(k) & \bar{L}_j(k+1)B(k) \end{bmatrix} \\ = \; & \begin{bmatrix} S & 0 \\ 0 & R \end{bmatrix} + \begin{bmatrix} A(k) & B(k) \end{bmatrix}^T \begin{bmatrix} I & \bar{L}_j^T(k+1) \end{bmatrix} \\ & \times \bar{H}_j(k+1)\begin{bmatrix} I & \bar{L}_j^T(k+1) \end{bmatrix}^T \begin{bmatrix} A(k) & B(k) \end{bmatrix} \end{aligned} \tag{25}$$

From Eq. (18), we have

$$\begin{aligned} & \bar{P}_j(k+1) \\ = \; & \begin{bmatrix} I & \bar{L}_j^T(k+1) \end{bmatrix} \bar{H}_j(k+1) \begin{bmatrix} I & \bar{L}_j^T(k+1) \end{bmatrix}^T \end{aligned} \tag{26}$$

where $\bar{P}_j(k+1)$ is the approximation of $P(k+1)$. Then, Eq. (25) becomes

$$\begin{aligned}&\bar{H}_{j+1}(k)\\ =\;&\begin{bmatrix}S & 0\\ 0 & R\end{bmatrix}+\begin{bmatrix}A(k) & B(k)\end{bmatrix}^T\bar{P}_j(k+1)\\ &\begin{bmatrix}A(k) & B(k)\end{bmatrix}\\ =\;&\begin{bmatrix}\bar{H}_{j+1,\xi\xi} & \bar{H}_{j+1,\xi u}\\ \bar{H}_{j+1,u\xi} & \bar{H}_{j+1,uu}\end{bmatrix}\end{aligned}\tag{27}$$

where

$$\begin{aligned}\bar{H}_{j+1,\xi\xi}&=A^T(k)\bar{P}_j(k+1)A(k)+S\\ \bar{H}_{j+1,\xi u}&=\bar{H}^T_{j+1,u\xi}=A^T(k)\bar{P}_j(k+1)B(k)\\ \bar{H}_{j+1,uu}&=B^T(k)\bar{P}_j(k+1)B(k)+R\end{aligned}\tag{28}$$

Similar to Eq. (26), we have

$$\bar{P}_{j+1}(k)=\begin{bmatrix}I & \bar{L}^T_{j+1}(k)\end{bmatrix}\bar{H}_{j+1}(k)\begin{bmatrix}I & \bar{L}^T_{j+1}(k)\end{bmatrix}^T\tag{29}$$

By substituting Eq. (27) into Eq. (29), the following equation can be obtained

$$\begin{aligned}\bar{P}_{j+1}(k)\;=\;&\begin{bmatrix}I & \bar{L}^T_{j+1}(k)\end{bmatrix}\begin{bmatrix}\bar{H}_{j+1,\xi\xi} & \bar{H}_{j+1,\xi u}\\ \bar{H}_{j+1,u\xi} & \bar{H}_{j+1,uu}\end{bmatrix}\\ &\times\begin{bmatrix}I & \bar{L}^T_{j+1}(k)\end{bmatrix}^T\end{aligned}\tag{30}$$

From Eq. (16), we have

$$\begin{aligned}\bar{L}_{j+1}(k)\;&=\;-\bar{H}^{-1}_{j+1,uu}\bar{H}_{j+1,u\xi}\\ &=\;-[B^T(k)\bar{P}_j(k+1)B(k)+R]^{-1}\\ &\quad\times[B^T(k)\bar{P}_{j+1}(k)A(k)]\end{aligned}\tag{31}$$

Substituting Eq. (31) into Eq. (30), we obtain

$$\begin{aligned}&\bar{P}_{j+1}(k)\\ =\;&A^T(k)\bar{P}_j(k+1)A(k)+S-A^T(k)\bar{P}_j(k+1)\\ &\times B(k)[R+B^T(k)\bar{P}_j(k+1)B(k)]^{-1}\\ &\times B^T(k)\bar{P}_j(k+1)A(k)\end{aligned}\tag{32}$$

Noticing that j is the policy iteration index and Eq. (32) is actually the DARE equation of the time-varying system described in Eq. (7), it can be concluded that $\bar{P}_{j+1}(k)$ will converge to $P(k)$ and $\bar{H}_{j+1}(k)$ will converge to $H(k)$. From the definition of Q-function in Eq. (19) and the control policy in Eq. (16), it can be concluded that $Q_j(\xi(k),u(k))$ will converge to $Q^*(\xi(k),u^*(k))$ and the control policy $v_{j+1}(\xi(k))$ will converge to $u^*(k)$ as $j\to\infty$. This completes the proof.

Remark 4: The system matrices $A(k)$ and $B(k)$ are only used for the convergence proof and their knowledge is not required in the following control design.

Lemma 2: For the Q-learning approximation described in Eqs. (21) and (22), if $\bar{H}_0(k+1)$ is chosen as a positive definite matrix, $\bar{H}_{j+1}(k)$ will always stay positive definite given a non-zero initial state.

Proof 2: According to Eq. (24), we have

$$\begin{aligned}&z^T(k)\bar{H}_{j+1}(k)z(k)\\ =\;&z^T(k+1)\bar{H}_j(k+1)z(k+1)\\ &+z^T(k)\begin{bmatrix}S & 0\\ 0 & R\end{bmatrix}z(k)\end{aligned}\tag{33}$$

In the following, we prove that $\bar{H}_{j+1}(k)>0$ through mathematical induction. In the case of $j=0$, we have

$$\begin{aligned}&z^T(k)\bar{H}_1(k)z(k)\\ =\;&z^T(k+1)\bar{H}_0(k+1)z(k+1)\\ &+z^T(k)\begin{bmatrix}S & 0\\ 0 & R\end{bmatrix}z(k)>0\end{aligned}\tag{34}$$

for $\forall z(k)\neq 0$, since $\bar{H}_0(k+1)$ is positive definite. In the case of $j>0$, if $\bar{H}_j(k+1)$ is a positive definite matrix, then $z^T(k)\bar{H}_{j+1}(k)z(k)>0$ for $\forall z(k)\neq 0$, by considering Eq. (33). It completes the proof.

In the following, we will show how to solve the Q-learning approximation problem discussed in (21) and (22) using a recursive time-varying least square method.

The existing Q-function $Q_{j+1}(\xi(k),u(k))$ from the k-th time slot to ∞ at the j-th iteration can be parameterized in the following form

$$\begin{aligned}&Q_{j+1}(\xi(k),u(k))\\ =\;&z^T(k)\bar{H}_{j+1}(k)z(k)\\ =\;&(z^T(k)\otimes z^T(k))\text{vec}(\bar{H}_{j+1}(k))\\ =\;&(\text{vec}(\bar{H}_{j+1}(k)))^T(z(k)\otimes z(k))\end{aligned}\tag{35}$$

where "vec$(\cdot)$" is the matrix stretch, and "$\otimes$" is the Kronecker product. Similarly, the Q-function from the $(k+1)$-th time slot to ∞ at the $(j+1)$-th iteration can be derived as

$$\begin{aligned}&Q_j(\xi(k+1),v_j(\xi(k+1)))\\ =\;&z^T(k+1)\bar{H}_j(k+1)z(k+1)\\ =\;&(z^T(k+1)\otimes z^T(k+1))\text{vec}(\bar{H}_j(k+1))\\ =\;&(\text{vec}(\bar{H}_j(k+1)))^T(z(k+1)\otimes z(k+1))\end{aligned}\tag{36}$$

If we define

$$\begin{aligned}\bar{h}_j(k+1)\;&=\;\text{vec}(\bar{H}_j(k+1))\\ \bar{h}_{j+1}(k)\;&=\;\text{vec}(\bar{H}_{j+1}(k))\end{aligned}\tag{37}$$

Eq. (21) can be rearranged in the following Linear-in-Parameters (LIP) form

$$\begin{aligned}&\bar{h}^T_{j+1}(k)(z^T(k)\otimes z^T(k))\\ =\;&g(\xi(k),u(k))+\bar{h}^T_j(k+1)(z^T(k+1)\otimes z^T(k+1))\\ =\;&\theta^T(k)\phi(k)\end{aligned}\tag{38}$$

where $\theta(k)=\bar{h}_{j+1}(k)$ is the vector of system parameter and $\phi(k)=z^T(k)\otimes z^T(k)$ is the regressor vector. The above equation is important as it allows us to optimize over the current control policy by working backward in time. The defined Q-function in Eq. (19) can be regarded as the desired target function that we need to approximate $V^*(\xi(k))$ in the least square sense.

In order to identify the time-varying parameter $\theta(k)$, the

Exponentially Weighted Recursive Least Squares (EWRLS) method discussed in [40] is implemented in this paper. The EWRLS method is employed to minimize the following block-wise Mean Squared Error (MSE)

$$E(\theta,k)=\frac{1}{2}\sum_{i=1}^{k}\lambda^{k-i}(d(i)-\theta^T(i)\phi(i))^2 \tag{39}$$

where λ is the forgetting factor that satisfies $0<\lambda<1$ and $d(i)=g(\xi(i),u(i))+\bar{h}_j^T(i+1)(z^T(i+1)\otimes z^T(i+1))$. A rule of thumb to choose λ is that λ with smaller values puts greater emphasis on the recent data.

The parameter $\theta(k)$ that minimizes Eq. (39) is given recursively by

$$\begin{aligned}\hat{\theta}(k+1) &= \hat{\theta}(k)+F(k+1)(d(k+1)\\ &\quad -\phi^T(k+1)\hat{\theta}(k))\end{aligned} \tag{40}$$

where $F(k)$ is the estimation gain matrix with

$$\begin{aligned}F(k+1) &= W(k+1)\phi(k+1)\\ &= W(k)\phi(k+1)(\lambda I\\ &\quad +\phi^T(k+1)W(k)\phi(k+1))^{-1}\\ W(k+1) &= \frac{1}{\lambda}(I-F(k+1)\phi^T(k+1))W(k)\end{aligned} \tag{41}$$

and $W(k)$ is the covariance matrix at the k-th time slot. To avoid $W(k)$ becoming too close to singularity, the covariance matrix is reset as follows

$$W(k)=\rho_0 I,\ \text{if}\ \sigma_{\min}(W(k))\le\rho_1 \tag{42}$$

where ρ_0 and ρ_1 are positive scalars and $\sigma(\cdot)$ denotes the eigenvalue of a matrix.

Remark 5: The covariance matrix $W(k)$ is reset by (42) to avoid the unlimited growth of the covariance matrix which may lead to large estimation errors. The same trick has been performed in many works on parameter estimation, which include [41], [42].

Remark 6: In the proposed method, we employ EWRLS to estimate time-varying parameters, which is widely acknowledged to exhibit fast convergence [40]. In a robotic interaction task, the parameter variation of a typical physical environment can be well handled by EWRLS. In very few cases, the problem of heavy computation burden may arise if the environment parameters change too fast, which needs to be further addressed.

The persistent excitation condition needs to be met to ensure the parameter convergence [43], [44]. Therefore, the exploration noise is added in the control input during the parameter estimation, i.e.,

$$u_e(k)=-L(k)\xi(k)+e(k) \tag{43}$$

where $e(0,\sigma^2)$ is a zero-mean white noise.

C. Optimal Impedance Adaptation

In this section, impedance adaptation will be developed based on the result in the previous subsection. We will first show how to transform a tracking problem into a regulation problem. Then, we will integrate the optimal critic learning discussed in Sections III-A and III-B into the impedance control in Section II. Under this adaptation, the target impedance function is adapted during the interaction process, which achieves an optimal performance.

For the damping-stiffness environment (1) described in Section II, the following cost function is considered

$$\begin{aligned}J_1(k) &= \sum_{k=1}^{\infty}[(x(k)-x_d(k))^T S_1(x(k)-x_d(k))\\ &\quad +f^T(k)R_1 f(k)]\end{aligned} \tag{44}$$

where S_1 is the weight of the trajectory tracking error, and R_1 is the weight of the interaction force. Besides, $S_1=S_1^T\ge 0$ and $R_1=R_1^T>0$.

As shown in Eq. (44), the optimal problem is in fact a tracking problem, which is concerned to make the robot arm follow or track a desired trajectory. However, the traditional optimal problem is usually a regulation problem which can be regarded as a special case where the desired trajectory is zero. Therefore, some manipulations are needed to make the problems identical. In particular, we consider

$$\eta(k)=[x^T(k)\ p^T(k)]^T \tag{45}$$

where $p(k)$ is the state of the following system

$$\begin{cases}p(k+1)=Up(k)\\ x_d(k)=Gp(k)\end{cases} \tag{46}$$

where U and G are two known matrices and (U,G) is observable.

Remark 7: Eq. (46) is to determine the desired trajectory $x_d(k)$ and provides the feasibility to employ the optimal control in the trajectory tracking problem. When U is not Hurwitz, Eq. (46) is able to generate a large variety of desired trajectories, including step, ramp, and others [6].

Considering the environment model (1), as the auxiliary state $p(k)$ is observable, the augmented matrices can be defined as follows

$$\begin{aligned}\bar{A}(k) &= \begin{bmatrix}A_e(k) & 0\\ 0 & U\end{bmatrix}, \bar{B}(k)=\begin{bmatrix}B_e(k)\\ 0\end{bmatrix},\\ \bar{S} &= \begin{bmatrix}S_1 & -S_1G\\ -G^TS_1 & G^TS_1G\end{bmatrix},\\ \bar{R} &= R_1\end{aligned} \tag{47}$$

Then, we have the augmented system

$$\eta(k+1)=\bar{A}(k)\eta(k)+\bar{B}(k)f(k) \tag{48}$$

and the corresponding cost function

$$J_1(k)=\sum_{k=1}^{\infty}(\eta^T(k)\bar{S}\eta(k)+f^T(k)\bar{R}f(k)) \tag{49}$$

The system described in Eq. (48) now has the same form as system (4) in Section III-A, so the optimal critic learning method can be adopted. It is trivial to show that the following

optimal control can be obtained

$$\begin{aligned} f(k) &= -K^*(k)\eta(k) \\ &= -K_1(k)x(k) - K_2(k)p(k) \\ &= -K_1(k)x(k) \\ &\quad -K_2(k)(G^T(k)G(k))^{-1}G^T(k)x_d(k) \end{aligned} \quad (50)$$

where $K^*(k)$ is equivalent to $L(k)$ in Eq. (16), and it is calculated using the method developed in Sections III-A and III-B. $K_1(k)$ and $K_2(k)$ are submatrices of $K^*(k)$. The exact impedance function $\psi(\cdot)$ in Eq. (2) which guarantees the optimal interaction is thus obtained.

We summarize the above procedures to compute the target impedance model, such that the desired interaction performance is achieved subject to unknown time-varying environments.

Algorithm 2 Optimal Impedance Adaptation

1: Choose an initial impedance model $\psi(0)$, and let $k = 0$.
2: Compute the inner-loop reference input $q_r(k)$ based on the impedance model (2), and apply a trajectory tracking method to make $\lim_{k\to\infty} q(k) = q_r(k)$.
3: Give the constructed state $p(k)$ in Eq. (46), and measure the interaction force $f(k)$ and position $x(k)$. Compute the utility function at the k-th step as below

$$g(\eta(k), f(k)) = \eta^T(k)\hat{S}\eta(k) + f^T(k)\hat{R}f(k) \quad (51)$$

4: Based on Eq. (40), apply EWRLS to estimate the optimal impedance parameter $\bar{H}_{j+1}(k)$ or $\bar{h}_{j+1}(k) = \text{vec}(\bar{H}_{j+1}(k))$.
5: Update the impedance model $\psi(k)$ as

$$f(k) = -\hat{H}_{ff}(k)^{-1}\hat{H}_{f\eta}(k)\eta(k) + e(k) \quad (52)$$

where $\hat{H}_{ff}(k)$ and $\hat{H}_{f\eta}(k)$ are submatrices of $\hat{H}(k)$ as in Eq. (14), and $\hat{H}(k)$ is the estimation of $\bar{H}_{j+1}(k)$.
6: Let $k \leftarrow k + 1$ and go to Step 2.

IV. Simulation Studies

To verify the proposed optimal critic learning for interaction control, in this section, a robot arm with two-degrees-of-freedom is considered to physically interact with an environment. The damping-stiffness environment model described by (1) is considered with the following parameters:

$$\begin{aligned} A_e(k) &= 1 - \frac{0.004}{0.1[\sin(5\times 10^{-4}k) + 1.1]} \\ B_e(k) &= -\frac{0.01}{0.1[\sin(5\times 10^{-4}k) + 1.1]} \end{aligned} \quad (53)$$

The parameters of the robot arm are given in Table I where m_j, l_j, I_j, $j = 1, 2$, represent the mass, the length, the inertia moment about the z-axis that comes out of the page passing through the center of mass, and the distance from the previous joint to the center of mass of the current link, respectively.

The initial coordinates of the robot arm in the joint space are given as $q_1(0) = \frac{\pi}{3}$ and $q_2(0) = -\frac{2\pi}{3}$, thus from the robot kinematics, the initial position in the Cartesian space is $x(0) = [0.4\ 0]^T$. The interaction force is only exerted to the robot arm along the x axis and the y axis is interaction-free. Adaptive control in [32] is adopted to guarantee the inner-loop control performance. The desired trajectory in the Cartesian space is determined by Eq. (46) with $U = 0.99$ and $G = 1$.

TABLE I
Parameters of the Robot Arm

Parameter	Description	Value
m_1	Mass of link 1	2.00kg
m_2	Mass of link 2	0.85kg
l_1	Length of link 1	0.40m
l_2	Length of link 2	0.40m
I_1	Inertia moment of link 1	0.02kgm^2
I_2	Inertia moment of link 2	0.02kgm^2

As the environment parameters $A_e(k)$ and $B_e(k)$ are known in simulation, the exact optimal solution (or desired impedance model) can be obtained by solving the DARE (7) which is referred to as "LQR". This desired impedance model is used to compare with the one obtained by the proposed method, which does not require the knowledge of the environment parameters.

A. Comparison: Proposed Method and LQR

In the first case, the weights in Eq. (44) are given by $S_1 = 1$ and $R_1 = 0.2$. The simulation results are shown in Figs. 2-4. In Fig. 2, the reference, desired, and actual trajectories are demonstrated and compared. From Fig. 2, it can be found that the trajectory using the proposed method converges to the desired one obtained by LQR. The tracking performance is not good at the initial stage, which is due to the fact that adaptation takes time. In practice, if we have some prior knowledge of the environment, better initial control parameters can be selected, which will help in improving the tracking performance at the initial stage. From Fig. 3, it is found that the interaction force under the proposed method also tracks the desired one under LQR. More details can be found in Fig. 4, where the convergence of control gains is illustrated. As discussed in Section III-C, the control gains are equivalent to desired impedance parameters, so the desired impedance model is obtained as in Fig. 4, which realizes the expected optimal interaction control.

To further illustrate the effectiveness of the proposed method, another two sets of cost functions are chosen in the second case. The weights in Eq. (44) are given by $S_1 = 1$, $R_1 = 0.01$ and $S_1 = 0.5$, $R_1 = 0.2$, respectively. The initial conditions are the same as above. Compared to the simulation results in the first case, if the weight of the tracking error is larger, it is expected that the tracking error becomes smaller and interaction force becomes larger. Conversely, if the weight of the tracking error is smaller, it is expected that the tracking error becomes larger and interaction force becomes smaller. The desired impedance model is again obtained based on known $A_e(k)$ and $B_e(k)$ for the comparison purpose. The simulation results in this case are given in Figs. 5-10, which are coherent with the expectations. It can be concluded

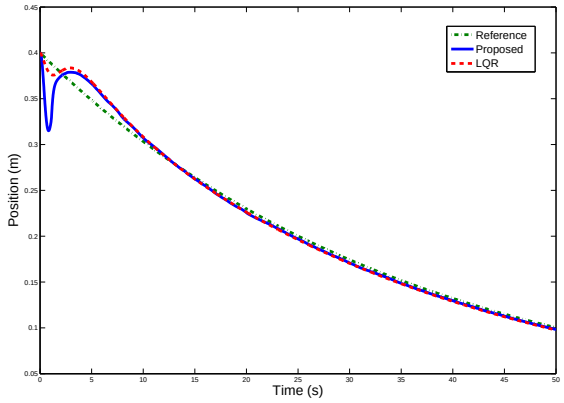

Fig. 2. Desired and actual trajectories, $S_1 = 1$ and $R_1 = 0.2$

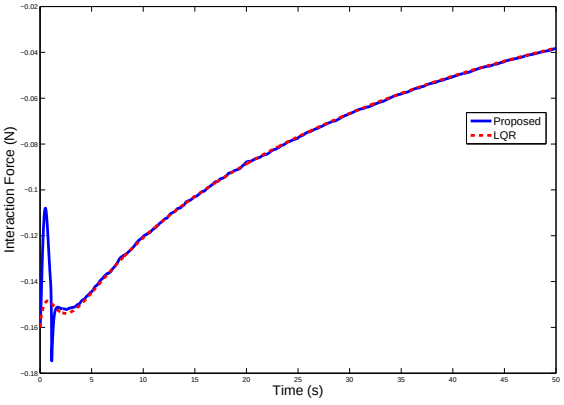

Fig. 3. Interaction forces, $S_1 = 1$ and $R_1 = 0.2$

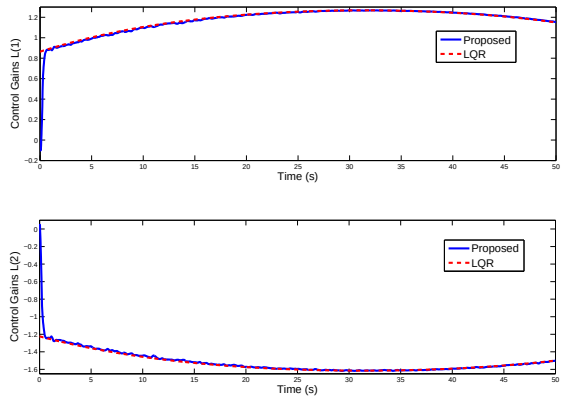

Fig. 4. Adaptation of impedance parameters, $S_1 = 1$ and $R_1 = 0.2$

that different S_1 and R_1 can be chosen to realize different interaction performances, e.g., either "softer" interaction or more accurate trajectory tracking.

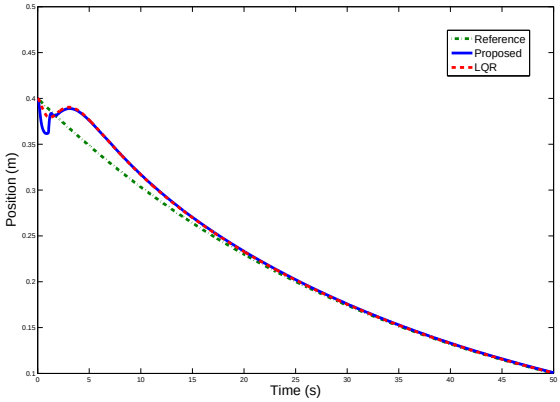

Fig. 5. Desired and actual trajectories, $S_1 = 1$ and $R_1 = 0.01$

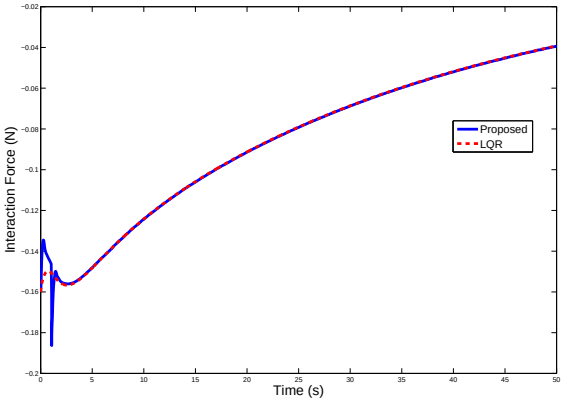

Fig. 6. Interaction forces, $S_1 = 1$ and $R_1 = 0.01$

B. Comparison: Proposed Method and Time-Invariant Method

As discussed in the Introduction, in the early works of impedance control, a set of desired constant impedance parameters is usually used [3]. In order to compare the performances of impedance control with or without impedance adaptation, additional simulation is conducted where the impedance parameters are fixed to desired values selected based on the known initial environment dynamics ("Time-Invariant"). The simulation results are shown in Figs. 11 and 12. It can be found that when the environment changes with respect to time, impedance parameters are no longer suitable to guarantee an optimal interaction performance. The trajectory using a fixed set of impedance parameters cannot track the desired trajectory under time-varying LQR. From the above comparison, it can be concluded that a good interaction performance cannot be guaranteed without the impedance adaptation in the case of time-varying environments.

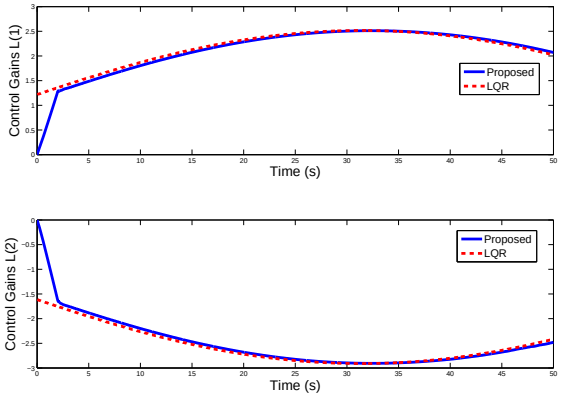

Fig. 7. Adaptation of impedance parameters, $S_1 = 1$ and $R_1 = 0.01$

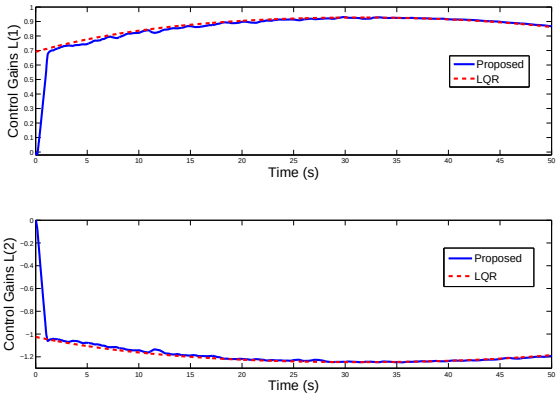

Fig. 10. Adaptation of impedance parameters, $S_1 = 0.5$ and $R_1 = 0.2$

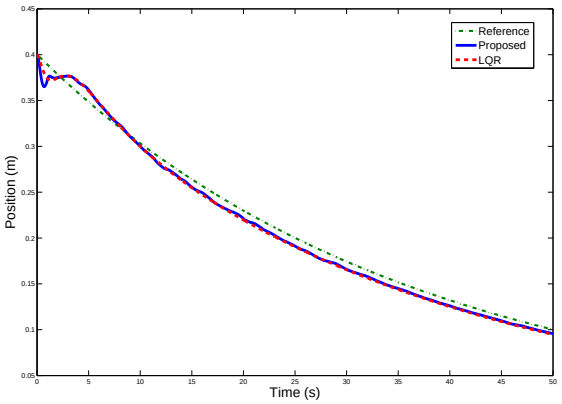

Fig. 8. Desired and actual trajectories, $S_1 = 0.5$ and $R_1 = 0.2$

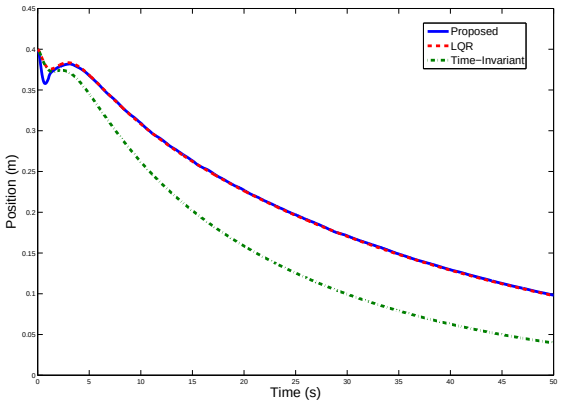

Fig. 11. Comparison of actual trajectories using different methods

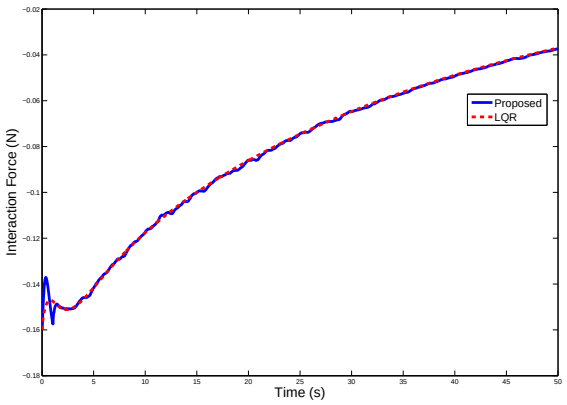

Fig. 9. Interaction forces, $S_1 = 0.5$ and $R_1 = 1$

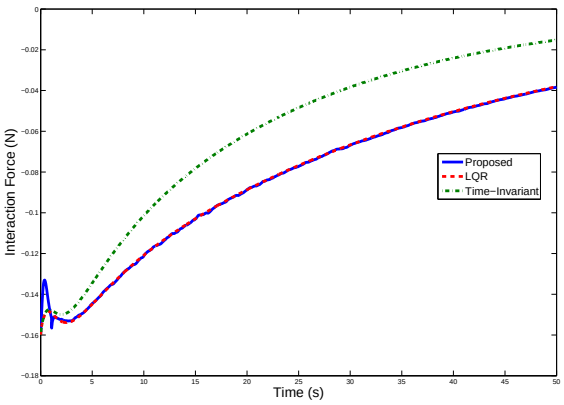

Fig. 12. Comparison of interaction forces using different methods

V. Conclusion

In this paper, we have proposed impedance adaptation for unknown and time-varying environments. In order to derive an optimal control for time-varying systems, a modified temporal difference equation has been employed and critic learning has been developed. This temporal difference equation was solved using a recursive time-varying least square method. Based on the proposed method, impedance adaptation for time-varying environments was realized, where optimal impedance parameters were obtained online without any prior knowledge of the environment dynamics. Simulation studies have been conducted to verify the feasibility of the proposed method.

References

[1] J. J. Craig and M. H. Raibert, "A systematic method of hybrid position/force control of a manipulator," *Computer Software and Applications Conference, IEEE Computer Society*, pp. 446–451, 1979.

[2] N. Hogan, "Impedance control: an approach to manipulation-part I: Theory; part II: Implementation; part III: Applications," *Transaction ASME J. Dynamic Systems, Measurement and Control*, vol. 107, no. 1, pp. 1–24, 1985.

[3] Y. Li, S. S. Ge, and C. Yang, "Learning impedance control for physical robot-environment interaction," *International Journal of Control*, vol. 85, no. 2, pp. 182–193, 2012.

[4] Y. Li and S.S. Ge, "Impedance Learning for Robots Interacting With Unknown Environments," *IEEE Transactions on Control System Technologies*, vol. 22, no. 4, pp. 1422–1432, 2014.

[5] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schaeffer, and E. Burdet, "Human-like adaptation of force and impedance in stable and unstable interactions," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 918–930, 2011.

[6] S. S. Ge, Y. Li, and C. Wang, "Impedance adaptation for optimal robot-environment interaction," *International Journal of Control*, vol. 87, no. 2, pp. 249–263, 2013.

[7] R. Johansson and M. W. Spong, "Quadratic optimization of impedance control," in *Proceedings of IEEE International Conference of Robotics and Automation*, vol. 1, pp. 616–621, 1994.

[8] M. Matinfar and K. Hashtrudi-Zaad, "Optimization-based robot compliance control: Geometric and linear quadratic approaches," *The International Journal of Robotics Research*, vol. 24, no. 8, pp. 645–656, 2005.

[9] L. J. Love and W. J. Book, "Environment estimation for enhanced impedance control," in *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1854–1859, IEEE, 1995.

[10] N. Diolaiti, C. Melchiorri, and S. Stramigioli, "Contact impedance estimation for robotic systems," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 925–935, 2005.

[11] D. Vrabie and F. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Networks*, vol. 22, no. 3, pp. 237–246, 2009.

[12] S. S. Ge, C. Yang, S.-L. Dai, Z. Jiao, and T. H. Lee, "Robust adaptive control of a class of nonlinear strict-feedback discrete-time systems with exact output tracking," *Automatica*, vol. 45, no. 11, pp. 2537–2545, 2009.

[13] P. J. Werbos, "A menu of designs for reinforcement learning over time," *Neural Networks for Control*, pp. 67–95, 1990.

[14] D. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1. Athena Scientific Belmont, MA, 1995.

[15] P. J. Werbos, "Intelligence in the brain: A theory of how it works and how to build it," *Neural Networks*, vol. 22, no. 3, pp. 200–212, 2009.

[16] F. Y. Wang, N. Jin, D. Liu, and Q. Wei, "Adaptive dynamic programming for finite-horizon optimal control of discrete-time nonlinear systems with ϵ error bound" *IEEE Transactions on Neural Networks*, vol. 22, no. 1, pp. 24–36, 2011.

[17] G. G. Lendaris, "Adaptive dynamic programming approach to experience-based systems identification and control," *Neural networks*, vol. 22, no. 5, p. 822, 2009.

[18] G. G. Lendaris, "Higher level application of ADP: A next phase for the control field?," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 4, pp. 901–912, 2008.

[19] P. J. Werbos, "Foreword-ADP: The key direction for future research in intelligent control and understanding brain intelligence," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 4, pp. 898–900, 2008.

[20] H. Zhang, Y. Luo, and D. Liu, "Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints," *IEEE Transactions on Neural Networks*, vol. 20, no. 9, pp. 1490–1503, 2009.

[21] A.-H. Tan, N. Lu, and D. Xiao, "Integrating temporal difference methods and self-organizing neural networks for reinforcement learning with delayed evaluative feedback," *IEEE Transactions on Neural Networks*, vol. 19, no. 2, pp. 230–244, 2008.

[22] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32–50, 2009.

[23] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers," *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 76–105, 2012.

[24] F. L. Lewis and D. Liu, *Reinforcement learning and approximate dynamic programming for feedback control*, vol. 17. John Wiley & Sons, 2013.

[25] B. Kim, J. Park, S. Park, and S. Kang, "Impedance learning for robotic contact tasks using natural actor-critic algorithm," *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, vol. 40, no. 2, pp. 433–443, 2010.

[26] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *International Journal of Robotics Research*, vol. 30, pp. 820–833, 2011.

[27] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems," *Proceedings of the IEEE Conference on Decision and Control*, vol. 23, pp. 1064–1069, 1984.

[28] M. Cohen and T. Flash, "Learning impedance parameters for robot control using an associative search network," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 382–390, June 1991.

[29] T. Tsuji and P. G. Morasso, "Neural Network learning of robot arm impedance in operational space," *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, vol. 26, no. 2, pp. 290–298, 1996.

[30] Y. Shmaliy, "Linear time-varying systems," in *Continuous-Time Systems*, pp. 349–423, Springer, 2007.

[31] G. Gilardi and I. Sharf, "Literature survey of contact dynamics modelling," *Mechanism and machine theory*, vol. 37, no. 10, pp. 1213–1239, 2002.

[32] J.-J. Slotine and W. Li, "On the adaptive control of robot manipulators," *International Journal of Robotics Research*, vol. 6, no. 3, pp. 147–157, 1987.

[33] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*. London: World Scientific, 1998.

[34] E. Burdet, G. Ganesh, C. Yang, and A. Albu-Schäffer, "Interaction force, impedance and trajectory adaptation: by humans, for robots," in *Experimental Robotics*, pp. 331–345, Springer, 2014.

[35] P. J. Werbos, "Consistency of HDP applied to a simple reinforcement learning problem," *Neural Networks*, vol. 3, no. 2, pp. 179–189, 1990.

[36] A. G. Barto, R. S. Sutton, and C. J. Watkins, "Learning and sequential decision making," in *Learning and computational neuroscience*, Citeseer, 1989.

[37] T. Landelius, "Reinforcement learning and distributed local model synthesis," 1997.

[38] P. Lancaster and L. Rodman, *Algebraic riccati equations*. Oxford University Press, 1995.

[39] C. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, Springer, 2009.

[40] K. J. Astrom and B. Wittenmark, *Adaptive Control*. Reading, Mass: Addison-Wesley, 1989.

[41] Y. Zhu and P. R. Pagilla, "Adaptive estimation of time-varying parameters in linear systems," in *Proceedings of the American Control Conference*, vol. 5, pp. 4167–4172, 2003.

[42] K. Xiong, H. Zhang, and L. Liu, "Adaptive robust extended Kalman filter for nonlinear stochastic systems," *IET Control Theory & Applications*, vol. 2, no. 3, pp. 239–250, 2008.

[43] S. S. Ge, "Adaptive controller design for flexible joint manipulators," *Automatica*, vol. 32, no. 2, pp. 273–278, 1996.

[44] T. Zhang, S. S. Ge, C. Hang, and T. Chai, "Adaptive control of first-order systems with nonlinear parameterization," *IEEE Transactions on Automatic Control*, vol. 45, no. 8, pp. 1512–1516, 2000.

Chen Wang received the B.Eng. degree in Control Science Engineering from Shandong University, China, in 2011. He is currently working towards the Ph.D. degree at the Department of Electrical and Computer Engineering, National University of Singapore. His current research interests include artificial intelligence, learning in human robot interaction, and human-robot collaboration.

Yanan Li (M'14) received the B.Eng degree in control science and engineering and the M.Eng degree in control and mechatronics engineering, from the Harbin Institute of Technology, China, in 2006 and 2009, respectively, and the Ph.D. degree from the NUS Graduate School for Integrative Sciences and Engineering, National University of Singapore, Singapore, in 2013. Currently, he is a Research Scientist at the Institute for Infocomm Research, Agency for Science, Technology and Research (A*STAR), Singapore. His research interests include physical human-robot interaction and human-robot collaboration.

Shuzhi Sam Ge (S'90-M'92-SM'99-F'06) received the B.Sc. degree from Beijing University of Aeronautics and Astronautics, Beijing, China, in 1986, and the Ph.D. degree from the Imperial College of Science, Technology and Medicine, University of London, London, U.K., in 1993.

He is the Founding Director of the Robotics Institute and the Institute of Intelligent Systems and Information Technology, University of Electronic Science and Technology of China, Chengdu, China. He is the Founding Director of the Social Robotics Laboratory, Interactive Digital Media Institute, National University of Singapore. He is a Professor in the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. He has authored or coauthored six books and more than 300 international journal and conference papers. His current research interests include social robotics, multimedia fusion, medical robots, and intelligent systems.

Dr. Ge is the Editor-in-Chief of the International Journal of Social Robotics. He has served/been serving as an Associate Editor for a number of flagship journals. He also serves as an Editor of the Taylor & Francis Automation and Control Engineering Series. He also served as the Vice President of Technical Activities, 2009-2010, and the Vice President for Membership Activities, 2011-2012, IEEE Control Systems Society.

Tong Heng Lee (M'90) received the B.A. degree (with first class honors) in engineering tripos from Cambridge University, Cambridge, U.K., in 1980 and the Ph.D. degree in electrical engineering from Yale University, New Haven, CT, in 1987.

He is a Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, and is currently the Head of the Drives, Power, and Control Systems Group in this department. He is the Deputy Editor-in-Chief of the International Federation of Automatic Control Mechatronics International Journal and serves as the Associate Editor of many other flagship journals. He has also coauthored three research monographs and holds four patents (two of which are in the technology area of adaptive systems, and the other two are in the area of intelligent mechatronics). His research interests are in the areas of adaptive systems, knowledge-based control, intelligent mechatronics, and computational intelligence.

Dr. Lee received the Cambridge University Charles Baker Prize in Engineering.